11. Wall-D

Program Name: Walld.java Input File: walld.dat

Wall-E's ancestor Wall-D was used when there was less trash. Wall-D would always try to get a specific amount of trash. You will help Wall-D find the length of the shortest path to obtain a certain amount of trash.
You will be given a square matrix (0 indexed) of size n x n. Wall-D starts at (n/2, n/2) (both floored). Wall-D must also pick up all trash he is ever on top of. He can move up, down, left, or right to find k trash. Each cell in the matrix is k, the amount of trash in that spot. Cells can be used multiple times and will have the same value each time they are used.

Input

The first line will contain the number of test cases T.
Each test case starts with a line containing two numbers. The first number is a side-length of the matrix n, and the next number is the amount of trash Wall-D wants to pick-up.
The next n lines show the matrix, with each cell separated by a space.

Output

Output the length of the shortest path (where 0 means that Wall-D starts on the desired amount of trash), or -1 if it is not possible to pick up exactly the required amount of trash.

Constraints

```
1 <= T <= 20
2 <= n <= 20
1 <= k <= 10^9
```

Each cell is a positive integer.

Example Input File

```
3
3 8
1 3 4
2 5 6
1 2 4
3 9
2 4 6
4 6 2
6 2 4
4 5
1 1 1 1
1 1 1 1
1 1 1 1
1 1 1 1
```

Example Output to Screen

```
1
-1
4
```

Explanation of the example

Start at 5 and move to 3; this sums to 8 and is a path length of 1. 5-2-1 also sums to 8, but is a longer path.
For the second case, the desired sum is odd, but all trash amounts are even, so it is impossible.